

**Copy/Move Graphical User Interface Apparatus  
and Method**

5

**BACKGROUND OF THE INVENTION**

**1. Technical Field:**

The present invention is directed to a copy/move  
10 graphical user interface (GUI) apparatus and method.  
Specifically, the invention is directed to a graphical  
user interface for informing a user of the progress of a  
copy/move operation and the functionality of allowing a  
user to modify the copy/move operation during operation  
15 execution.

**2. Description of Related Art:**

The ability to copy and/or move files from one  
location in a computer system to another location is  
generally known in the art. When a copy operation, for  
20 example, is performed using known devices, a copy status  
graphical user interface is commonly displayed showing  
the name of the current file that is being copied and,  
optionally, a progress status bar. The progress status  
bar indicates graphically, the portion of the current  
25 file that has been copied and the portion that has not  
been copied. The progress status bar "fills up" from  
left to right as more of the file is copied from one  
location to another. When copying of a current file is  
complete, the same copy status graphical user interface  
30 is displayed for the next file in a series of files that  
is to be copied.

The copy status graphical user interface may further  
include a virtual button for canceling the copy  
operation. If a user selects the virtual cancel button,

Docket No. AUS990915US1

the entire copy operation is stopped. Consequently, the current file and any other files not already copied will not be copied.

Thus, with the known devices, the copy graphical user interface only informs a user of the current file being copied and does not allow a user to make modifications to the copy operation while the operation is being performed. As the present inventors have recognized, it would be advantageous to have an apparatus and method for providing a copy/move graphical user interface that informs the user of the current file status, the status of other files that are to be copied/moved in the copy/move operation, and that allows a user to make modifications to the copy/move operation while the copy/move operation is being performed.

**SUMMARY OF THE INVENTION**

5       The invention provides a copy/move graphical user interface apparatus and method. With the present invention, a copy/move instruction is received from a user via an input device. The received copy/move instruction indicates the files that are to be copied and  
10   the destination location of the copied files. In response to receiving the copy/move instruction, the identified files are placed in a copy/move queue. The total data size for the complete copy/move operation is then computed as the sum of the file sizes for all of the  
15   files that are to be copied/moved.

      Either before, after, or at the same time as the calculation of the total data size, a status graphical user interface (GUI) is displayed. The GUI includes information pertaining to the progress of the copy/move  
20   process for a particular file, the progress of the entire copy/move operation, a listing of files and their associated attributes that are to be copied/moved in the present copy/move operation, estimated times of  
25   completion, and virtual buttons and boxes for modifying the copy/move operation or the view of the GUI.

      Once the GUI is displayed, the copy/move operation is initiated with the first file in the copy/move queue. The copy/move process begins by first determining if the file has been designated by the user to be skipped. A  
30   user may enter, via an input device, a command for the copy/move operation to skip a particular file that is to

be copied/moved. The command may be entered during the copy/move operation but before the file is copied/moved. If a skip command has been entered for the file, the copying/moving of the file is not performed and the  
5 copy/move operation proceeds to the next file in the copy/move queue.

If the skip command is entered after a file has been copied/moved, the skip command will be interpreted as a delete command requesting that the designated file be  
10 deleted from the copy/move destination location.

If a skip command has not been entered for the current file, the file data is then copied/moved from the current file to a destination location. During the copying/moving of data from the current file to the  
15 destination location, a copy/move rate of the copy/move operation is measured. A current file size is also determined and used, along with the copy/move rate and the total data size for the copy/move operation, to  
20 dynamically update estimated times of completion for one or both of the current file copy/move operation and the total copy/move operation.

In addition, the amount of data that has been copied is determined and used, along with the current file size and the total data size for the copy/move operation, to  
25 update progress bars associated with the current file copy/move operation and the total copy/move operation.

Additionally, the user may change the order in which the GUI displays the files in the copy/move queue or the order in which the files are actually copied/moved in the  
30 copy/move queue. Thus, with the present invention, a user is informed of the progress of a copy/move operation by providing information pertaining to files that have

Docket No. AUS990915US1

already been copied/moved in the copy/move operation, and also information pertaining to the particular file being copied/moved, the files that are pending in the copy/move queue, the estimated time of completion, and a progress  
5 bar graphically depicting the portion of the file and the portion of the entire copy/move operation completed. In addition, the user may modify the copy/move operation by instructing the processor to skip files, delete files, and change the order in which files are copied/moved.

**BRIEF DESCRIPTION OF THE DRAWINGS**

5       The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself, however, as well as a preferred mode of use, further objectives and advantages thereof, will best be understood by reference to the following detailed  
10 description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

**Figure 1** is an exemplary diagram of a data processing apparatus in which the present invention may be implemented;

15       **Figure 2** is an exemplary block diagram of a processing system in which the present invention may be implemented;

**Figure 3** is an exemplary diagram of a copy/move status graphical user interface according to the present  
20 invention;

**Figure 4** is a flowchart outlining an exemplary operation for performing the copy/move operation according to the present invention; and

**Figure 5** is a flowchart outlining an exemplary  
25 operation for modifying a copy/move operation according to the present invention.

**DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT**

5       **Figure 1** is an exemplary diagram of a data processing apparatus in which the present invention may be implemented. The particular data processing apparatus shown in **Figure 1** is a computer **100** however, other data processing apparatus, such as networked workstations,  
10 network server apparatus, and the like may be used as the data processing apparatus of the present invention. The computer **100** is used with the present invention to provide a copy/move graphical user interface, such as the graphical user interface depicted in **Figure 3**.

15       The computer **100** may include a system unit **110**, a video display terminal **102**, a keyboard **104**, storage devices **108**, which may include floppy drives and other types of permanent and removable storage media, and pointing device **106**, such as a computer mouse.  
20 Additional input devices may be included with the computer **100**, such as a joystick, gamepad, touchpad, touchscreen, trackball, microphone, and the like. Computer **100** also preferably includes a graphical user interface that may be implemented by means of systems  
25 software residing in computer readable media in operation within computer **100**.

      With reference now to **Figure 2**, a block diagram illustrates a data processing system **200** in which the present invention may be implemented. Data processing  
30 system **200** is an example of a computer, such as computer **100** in **Figure 1**, in which code or instructions

implementing the processes of the present invention may be located. Data processing system **200** employs a peripheral component interconnect (PCI) local bus architecture.

Although the depicted example employs a PCI bus, other bus architectures such as Accelerated Graphics Port (AGP) and  
5 Industry Standard Architecture (ISA) may be used.

Processor **202** and main memory **204** are connected to PCI local bus **206** through PCI bridge **208**. PCI bridge **208** also may include an integrated memory controller and cache  
10 memory for processor **202**. Additional connections to PCI local bus **206** may be made through direct component interconnection or through add-in boards. In the depicted example, local area network (LAN) adapter **210**, small computer system interface (SCSI) host bus adapter **212**, and  
15 expansion bus interface **214** are connected to PCI local bus **206** by direct component connection. In contrast, audio adapter **216**, graphics adapter **218**, and audio/video adapter **219** are connected to PCI local bus **206** by add-in boards inserted into expansion slots. Expansion bus interface  
20 **214** provides a connection for a keyboard and mouse adapter **220**, modem **222**, and additional memory **224**. SCSI host bus adapter **212** provides a connection for hard disk drive **226**, tape drive **228**, and CD-ROM drive **230**. Typical PCI local bus implementations will support three or four PCI  
25 expansion slots or add-in connectors.

An operating system runs on processor **202** and is used to coordinate and provide control of various components within data processing system **200** in **Figure 2**. The operating system may be a commercially available operating  
30 system, such as OS/2 which is available from International



Business Machines Corporation. "OS/2" is a trademark of International Business Machines Corporation. Instructions for the operating system, the object-oriented operating system, and applications or programs are located on  
5 storage devices, such as hard disk drive **226**, and may be loaded into main memory **204** for execution by processor **202**.

Those of ordinary skill in the art will appreciate that the hardware in **Figure 2** may vary depending on the  
10 implementation. Other internal hardware or peripheral devices, such as flash ROM (or equivalent nonvolatile memory) or optical disk drives and the like, may be used in addition to or in place of the hardware depicted in **Figure 2**. Also, the processes of the present invention  
15 may be applied to a multiprocessor data processing system.

For example, data processing system **200**, if optionally configured as a network computer, may not include SCSI host bus adapter **212**, hard disk drive **226**,  
20 tape drive **228**, and CD-ROM **230**, as noted by dotted line **232** in **Figure 2** denoting optional inclusion. In that case, the computer, to be properly called a client computer, must include some type of network communication interface, such as LAN adapter **210**, modem **222**, or the  
25 like. As another example, data processing system **200** may be a stand-alone system configured to be bootable without relying on some type of network communication interface, whether or not data processing system **200** comprises some type of network communication interface. As a further  
30 example, data processing system **200** may be a Personal Digital Assistant (PDA) device which is configured with

ROM and/or flash ROM in order to provide non-volatile memory for storing operating system files and/or user-generated data.

5 The depicted example in **Figure 2** and above-described examples are not meant to imply architectural limitations. For example, data processing system **200** also may be a notebook computer or hand held computer in addition to taking the form of a PDA. Data processing system **200** also may be a kiosk or a Web appliance.

10 The processor **202** performs the functions described herein, based on instructions obtained from a user and computer program instructions stored, for example, in main memory **204**, in memory **224**, or instructions contained on one or more peripheral devices **226-230**. In a  
15 preferred embodiment, the processor **202** performs the functions in a Microsoft Windows™ operating environment, although other windowed or non-windowed operating environments may be utilized without departing from the spirit and scope of the present invention.

20 The processor **202** receives a copy/move instruction from a user via the pointing device **106**, keyboard **104**, or the like, and keyboard and mouse adapter **220**. A copy instruction is an instruction to create a copy of an existing file and place it in a designated location.  
25 Typically, the copy instruction may be input by selecting files to be copied and then selecting a destination location to which the files are to be copied.

Alternatively, the copy instruction may be received from an executing computer program, such as an  
30 installation program for installing a computer program. In such a case, the files to be copied are identified by

the installation program.

Regardless of whether the copy instruction is received from a user or an executing computer program, the principles of this invention equally apply. The  
5 initial order in which files are copied to the destination location is set based on either the order in which the user selects the files to be copied or an order set by the executing computer program (e.g. an install program for installing a computer application, and the  
10 like). The following description of the invention will assume that the copy instruction is received from a user.

The selection of the files may be performed, for example, by highlighting a first file using a mouse **106**, pressing a "shift" key on the keyboard **104** and  
15 highlighting additional files to be copied. The destination location may be selected by, for example, dragging the highlighted files to a particular directory on a hard disk or peripheral device. Such copying instructions are generally known to those of ordinary  
20 skill in the art.

Similarly, a move instruction is a copy instruction in which the original file is deleted. Thus, for purposes of the description of this invention, a copy instruction will be assumed. However, the details of  
25 this invention are equally applicable to a move instruction.

The actual functions of copying or moving files from one directory to another, or from one computer resource, network resource, or the like, to another are widely  
30 known in the art. Operating systems, such as DOS, Windows, and the like make use of copy and move functions

on a routine basis. Therefore, the actual copy/move function will not be discussed herein in detail.

The copy instruction received by the processor **202** indicates the files that are to be copied and the  
5 destination location of the copied files. In response to receiving the copy instruction, the processor **202** places the identified files in a copy queue in main memory **204**, for example. The processor **202** calculates the total data  
10 size is the sum of the file sizes for the files that are to be copied. Thus, if three files having respective sizes, 100KB, 200KB and 300KB, are to be copied, the total data size for the copy operation will be 600KB.

The processor **202** also instructs the graphics  
15 adapter **218** to create a copy status graphical user interface, hereafter referred to as the GUI, which is then displayed by a display device, such as display **102**, via the audio/video adapter **219**. The GUI includes information pertaining to the progress of the copy  
20 process for a particular file, the progress of the entire copy operation, a listing of files and their attributes that have been copied, are being copied, and are pending to be copied in the present copy operation, estimated remaining times for completion, and virtual buttons and  
25 boxes for modifying the copy operation or the view of the GUI.

Once the GUI is displayed, the copy operation is initiated with the first file in the copy queue. The processor **202** begins the copy process by first  
30 determining if this file has been designated by the user to be skipped. A user may enter, via the keyboard **104**,

Docket No. AUS990915US1

mouse **106**, or other input device, a command for the copy operation to skip a particular file that is to be copied. The command may be entered prior to initiation of the copy operation or during the copy operation but before  
5 the file is copied. If a skip command has been entered for the file, the copying of the file is not performed and the copy operation proceeds to the next file in the copy queue.

If the skip command is entered after a file has been  
10 copied, the skip command will be interpreted by the processor **202** as a delete command requesting that the designated file that was previously copied be deleted from the destination location. The deletion of a file from the destination location will be described in more  
15 detail hereafter.

If a skip command has not been entered for the current file, the processor **202** begins to copy data from the current file to a destination location. The destination location may be on an internal or external  
20 device. For example, the copy operation may copy files from an internal hard drive to a storage medium in a floppy drive, CD-ROM drive, ZIP drive, magnetic tape drive or other external storage medium.

The processor **202**, during the copying of data from  
25 the current file to the destination location, measures a copy rate of the copy operation. The copy rate may be, for example, in bits per second, bytes per second, or the like. The processor **202** also determines the current file size and uses this information along with the copy rate  
30 and the total data size for the copy operation, to update estimated times of completion for one or both of the

current file copy operation and the total copy operation.

In addition, the processor **202** determines the amount of data that has been copied and uses this information along with the current file size and the total data size for the copy operation to update progress bars associated with the current file copy operation and the total copy operation. If a complete file has been copied, the displayed listing of files in the queue is modified to designate that the file has been copied.

10 Once the entire copy operation has been completed, those files that have been marked as delete will be deleted from the destination location. A confirmation message may be displayed to the user so that the user may select a virtual button authorizing the deletion of the marked files or abort the deletion of these files.

15 Furthermore, during the copy operation, a user may enter a cancel command in order to cancel the entire copy operation. If the cancel command is entered, all files still pending to be copied will not be copied, a current copy operation is aborted, and all the files are removed from the copy queue. The cancel command may be entered by the user via any type of input device, such as pressing an appropriate key or combination of keys on a keyboard, selecting a virtual "cancel" button via a pointing device, entering a verbal command via a voice recognition device, and the like.

20 Additionally, the user may change the order in which the GUI displays the files in the copy queue or the order in which the files are actually copied in the copy queue. For example, a user may instruct the GUI to display the files in alphabetical order, by increasing or decreasing size, by date of creation, and the like. The user may

change the order in which the files will be copied by rearranging the displayed listing of files in the copy queue.

Thus, with the present invention, a user is informed  
5 of the progress of a copy/move operation by providing information pertaining to the particular file being copied, the files that have been copied, the files that are pending in the copy queue, the estimated time of completion, and a progress bar graphically depicting the  
10 portion of the file currently being copied and the portion of the entire copy operation completed. In addition, the user may modify the copy operation by instructing the processor to skip files, delete files, and change the order in which files are copied.

15 **Figure 3** is an exemplary diagram of a copy status GUI **300** according to the present invention. In this example, the GUI **300** is present in response to a copy instruction or command. This copy instruction may be administered by a command line interface or a GUI. With  
20 a GUI, a program such as Windows Explorer in Windows 98 may be used to select and copy multiple files.

As shown in **Figure 3**, the GUI **300** includes a field  
25 **305** for identifying the filename of the current file being copied and the source and destination paths from and to which the file is being copied. The GUI **300** further includes a current file progress bar **310**, an estimated time **315** remaining for the copying of the current file, an estimated time **320** for completion of the entire copy operation, a graphical icon illustrating the  
30 copy function, a virtual cancel button **330**, a copy session (or operation) progress bar **335**, and a copy queue

display **340**.

In this example, the graphical icon **325** may be stagnant or animated. The progress bars **310** and **335** "fill up" as the copy operation progresses. The virtual  
5 cancel button **330** may be selected in order to cancel the entire copy operation.

The copy session progress bar **335** may further include an indication of which file in the queue is currently being copied, such as "copying file 5 of 7."  
10 The estimated remaining times **315** and **320** are updated as the copy operation progresses, thereby decreasing as more data of the files is copied to the destination location. The estimated remaining times **315** and **320** may increase if, during a previous estimation, the copy rate was  
15 higher than the current copy rate.

The copy queue display **340** includes header buttons, such as "#", "Name", "Size", "Skip/Remove File", and the like, designating fields of the copy queue display for the various file attributes. The copy queue display **340**  
20 further includes a listing of files that are in the copy queue along with their respective attributes, in accordance with the header buttons. For example, each file in the copy queue is displayed with a copy order number, the filename of the file along with an indicator  
25 of whether the file has been copied or not, a file size, a skip/remove file indicator, and other file attributes, such as date of creation, and the like.

As files are copied to the designated destination location, the copy indicator is changed from a "not  
30 copied" indicator to a "copied" indicator. In the particular example shown in **Figure 3**, the "not copied"



Docket No. AUS990915US1

indicator is an empty box while the "copied" indicator is a box with a check mark.

Similarly, the skip/remove file indicator for each file is either a "do not skip/remove" indicator or a  
5 "skip/remove" indicator. In the example shown in **Figure 3**, the "do not skip/remove" indicator is an empty box while the "skip/remove" indicator is a box with an "X" and an associated text message that indicates whether the file will be skipped or deleted.

10 When a user selects the "do not skip/remove" indicator for a file that has not yet been copied, the "do not skip/remove" indicator will be changed to a "skip" indicator. Thus, the file will be skipped when the copy operation progresses to this file. If a user  
15 selects the "do not skip/remove" indicator for a file that has already been copied, the indicator will be changed to a "delete" indicator. As a result, the file will be deleted from the destination location when the total copy operation is completed. The skip or delete  
20 status of a file may be removed by again selecting the skip/remove indicator.

The header buttons of the copy queue display **340** may be selected in a first mode to change the view of the copy queue display **340** or in a second mode to change the  
25 order in which the files are copied. For example, if the "Name" header button is selected with a pointing device, such as a mouse, with a left mouse button, the listing of files in the copy queue display **340** may be changed so that the listing is in alphabetical order. If the "Name"  
30 header is again selected with the left mouse button, the listing may be changed to be in reverse alphabetical

Docket No. AUS990915US1

order. The same changes in view may be performed with file size (in order of increasing or decreasing size), date of creation (most recent to oldest or vice versa), and the like. Changing the view of the copy queue

5 display **340** does not change the order in which files are copied to the destination location.

Furthermore, the header buttons may be selected in a second mode, such as by a right mouse button, to change the order in which the files are copied to the  
10 destination location. For example, if the "Size" header is selected using the right mouse button, the order in which the files are copied will be changed to an order of smallest file size to largest file size. If the "Size" header is again selected using the right mouse button,  
15 the order will be changed to largest file size to smallest file size. The reordering only affects those files that have not already been copied to the destination location. The view of the copy queue display **340** may be modified accordingly to reflect the change in  
20 the copy order.

In addition to changing the copy order using the header buttons, the copy order may be changed by selecting a file listing in the copy queue display **340** and then selecting a new position for the file listing in  
25 the copy queue display **340**. For example, in the example shown in **Figure 3**, the file "file6.cpp" may be selected using a mouse and dragged to a position before "file5.exe." Thus, the copy order of the non-copied files in the copy queue will be changed such that  
30 "file6.cpp" will be copied before "file5.exe." This ability to change the copy order may be advantageous if

the user determines that he/she does not want to wait for "file5.exe," which is a very large file, to be copied before the relatively smaller file "file6.cpp" is copied.

Thus, the present invention provides the user with  
5 many options for modifying a copy operation while the copy operation is being executed. Furthermore, the present invention provides the user with detailed information about the copy operation so that the user may make informed decisions as to whether the copy operation  
10 satisfies the user's needs or whether the copy operation needs to be modified to suit the user's needs.

**Figure 4** is a flowchart outlining an exemplary operation for performing a copy operation in accordance with the present invention. The operation starts with a  
15 copy/move instruction being received from a user (step **405**). A list of the files that are to be copied/moved is then retrieved from the copy/move queue (step **410**). A total data size for the copy/move operation is calculated (step **415**) and the copy/move GUI is displayed (step **420**).

20 The copy/move operation for the next file in the copy/move queue is started (step **425**) and it is determined whether or not the file is to be skipped (step **430**). If the file is to be skipped (step **430:YES**), the operation skips the file and starts the copy/move  
25 operation for the next file in the copy/move queue (step **425**). If the file is not to be skipped (step **430:NO**), the file data is copied to the selected destination location (step **435**). The rate at which the data is copied/moved is determined (step **440**) and the estimated  
30 times remaining in the copying of the current file and the entire copy operation are updated (step **450**). The

Docket No. AUS990915US1

GUI may then be updated to reflect the updated estimated remaining times.

The amount of data that has been copied/moved is then determined (step **455**) and the progress bars for the  
5 current file and the copy operation are updated accordingly (step **460**). It is then determined whether the copy operation has been canceled by the user inputting a cancel command (step **465**). If there has been a cancel command entered (step **465:YES**), the copy/move  
10 operation is stopped (step **470**) and the operation ends (step **495**). If the cancel command has not been entered (step **465:NO**), it is determined whether or not the entire current file has been copied/moved (step **475**). If the current file has not been completely copied/moved (step  
15 **475:NO**), the operation continues to copy/move data to the destination location (step **435**). If the current file has been completely copied/moved (step **475:YES**), a determination is made as to whether the entire copy/move operation has been completed (step **480**).  
20 If the entire copy/move operation has not been completed (step **480:NO**), the copying/moving operation proceeds to the next file in the copy/move queue (step **425**). If the entire copy/move operation has been completed (step **480:YES**), it is determined whether any of  
25 the files copied were designated "delete" by the user during the copy operation (step **485**). If so (step **485:YES**), the files designated "delete" are deleted from the destination location (step **490**). This may occur after first asking the user to confirm the deletion of  
30 the files. If no files are designated "delete" (step

485:NO), the copy/move operation ends (step 495).

Figure 5 is a flowchart outlining an exemplary operation for modifying the copy order in a copy queue according to the invention. The operation shown in

5 Figure 5 is equally applicable to a move operation.

As shown in Figure 5, the operation starts with receiving a reorder instruction from a user (step 510). As described above, this reorder instruction may be from a second mode selection of a header button or from moving queue items within the copy/move queue. A determination is then made as to whether the files to be reordered have not been copied or are not currently being copied (step 520). If the files have already been copied or are currently being copied (step 530:YES), an error message is sent to the user (step 540) and the reorder operation is ended (step 570).

If the files have not already been copied or are not currently being copied (step 530:NO), the files are reordered according to the particular reorder instruction received (step 550), i.e. by file size, file name, particular position in the copy queue, and the like. The displayed copy queue is then updated based on the reordering of the files (step 560) and the operations ends (step 570).

25 Thus, with the present invention, a user is provided with all the necessary information to monitor and modify a copy/move operation while the copy/move operation is being performed. In this way, the user is provided with greater control over the copy/move operation than was previously possible with the known prior art.

30 It is important to note that while the present

invention has been described in the context of a fully functioning data processing system, those of ordinary skill in the art will appreciate that the processes of the present invention are capable of being distributed in the form of a computer readable medium of instructions and a variety of forms and that the present invention applies equally regardless of the particular type of signal bearing media actually used to carry out the distribution. Examples of computer readable media include recordable-type media such a floppy disc, a hard disk drive, a RAM, and CD-ROMs and transmission-type media such as digital and analog communications links.

The description of the present invention has been presented for purposes of illustration and description, but is not intended to be exhaustive or limited to the invention in the form disclosed. Many modifications and variations will be apparent to those of ordinary skill in the art. For example, although **Figure 3** depicts a particular format, the GUI of the present invention may vary in presentation. For example, estimated times **315** and **320** may be present using dials rather than text and progress bars **310** and **335** may be vertical bars that fill up as the copy operation progresses or may be circles that fill up radially as the copy operation progresses. Other additional information, such as verification operations involved in copying files, also may be included on the GUI **300**. A location of the destination file on the storage device also may be illustrated. Many other modifications to the GUI **300** may be made without departing from the spirit and scope of the present invention.

Docket No. AUS990915US1

5